Operating System

Planning Migration from Windows NT to Windows 2000

White Paper

Abstract

This paper outlines the planning processes and considerations when migrating Microsoft® Windows NT® domains to Microsoft Windows® 2000 and the Active Directory™ service. New Windows 2000 utilities, tools, and technologies make migrating users and computers, while maintaining access to resources, a straightforward task.

© 2000 Microsoft Corporation. All rights reserved.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Microsoft, Active Desktop, BackOffice, the BackOffice logo, MSN, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Other product and company names mentioned herein may be the trademarks of their respective owners.

Microsoft Corporation • One Microsoft Way • Redmond, WA 98052-6399 • USA

0399

CONTENTS

INTRODUCTION.....1

DETERMINING YOUR MIGRATION ROADMAP.....2
Migration Goals 2
Migration Concepts 4
Domain Upgrade 4
Implications of Domain Upgrade 5
Domain Restructure 26
When Should You Restructure? 27
Why Would You Restructure? 28
Implications of Domain Restructure 29

DECISION POINT CHECKLIST.....44
Upgrade Decisions 44
Restructure Decisions 45

SUMMARY.....46
For More Information 46

INTRODUCTION TO YOUR MIGRATION ROADMAP

This white paper is intended to guide you through planning a migration from Windows NT 3.51 and Windows NT 4.0 to Windows 2000 in an enterprise environment.

The focus of this white paper is on planning an implementation of the Active Directory namespace through the upgrade of Windows NT domains; most of the Active Directory namespace planning should have already taken place.

During any planning process, you make important choices. In planning your migration, such choices will determine the availability of certain desired features of the operating system.

This section helps you plan your priorities by covering:

-
- Concepts behind migration, so that you can understand the choices you make in formulating your migration roadmap, and the impact these might have on your migration goals

By the end of this section you should have enough information to complete your migration roadmap, and identify some areas of risk to your project.

Migration Concepts

There are two ways of arriving at the desired infrastructure:

- Domain upgrade – sometimes referred to as *in-place upgrade*
- Domain restructure – sometimes referred to as *domain consolidation* or *domain collapse*

Upgrade and restructure are not mutually exclusive; some organizations might first upgrade and then restructure, while others might restructure at the beginning.

Domain Upgrade

Upgrade to Windows 2000 Server is the easiest, least-risk migration route.

We can define domain upgrade as the process of upgrading the software on the Primary Domain Controller (PDC) of a domain, and upgrading some or all of the Backup Domain Controllers (BDCs), from Windows NT 4.0 to Windows 2000 Server.

Because Windows 2000 is designed to support mixed networks containing Windows 9*x*, Windows NT 4.0, and Windows 2000 with full interoperability, not all systems in the domain have to be upgraded to take advantage of Windows 2000 features. Having said this, upgrading

the PDC should be seen as only the first stage of the upgrade, since further incremental benefits are obtained by subsequently upgrading Backup Domain Controllers (BDCs), then Member Servers.

Because this is an operating system upgrade rather than a fresh installation, the existing domain structure, users, or groups are maintained, though in the process new Windows 2000 features are enabled. In fact, the biggest distinction between upgrade and consolidation lies in the fact that in upgrading we maintain the existing domain structure.

When you have completed your upgrade, and have access to advanced Windows 2000 management tools and features, you may wish to restructure your domains.You should be aware that restructuring is not a trivial task, and will require additional planning and implementation, as discussed in the section on restructure, later in this paper.

If structural change is one of your main goals of migration, you might wish to consider an approach where restructuring takes place during the migration.

Implications of Domain Upgrade

Upgrade represents the easiest, least-risk migration route because it retains most of your system settings, preferences, and program installations. In this section we will look at the actual effects of the upgrade on DCs and security principals, and explain the importance of these in making your planning decisions.

The Active Directory Installation Wizard (DCPromo.exe)

After forcing a synchronization of all BDCs in the domain, so that the BDCs are completely updated with any recent changes made at the PDC, we can start the account domain upgrade by upgrading the PDC.

After the installation of the core operating system on the PDC, the setup program detects that a DC is being upgraded, and prompts the administrator to install the Active Directory on the server by automatically running the DCPROMO program.

DCPROMO gives the administrator the option of creating the first tree in a new forest, creating a new tree in an existing forest, creating a replica of an existing domain, or installing a child domain. The choice you make will depend on the outcome of the namespace planning process.

Note: This is not intended as a detailed description of the process. You will also need to consider activities such backing up the domain in case you encounter serious problems and need to fall back. You might choose to add an additional BDC to the domain prior to the synchronization and upgrade, and then to take this BDC offline for the duration of the upgrade. More details can be found in the Deployment Planning Guide.

Windows NT PDC

As part of the Active Directory installation process, the contents of the Windows NT account database and the Security Account Manager (SAM) are copied into Active Directory. These objects are the security principals (user accounts, local and global groups, and computer accounts).

As soon as the PDC has been upgraded and Active Directory has been installed, the domain is running in Windows 2000 *mixed mode*. For a fuller description of mixed mode and native mode see the section “Mixed Mode and Native Mode,” below.

From this point on, the Windows 2000 Server PDC uses Active Directory to store objects. It is fully backward compatible because it exposes the data as a flat store to Windows NT BDCs during replication. This manifests itself in a number of ways:

- The PDC appears as a Windows 2000 DC to other Windows 2000 computers, and as a Windows NT PDC to computers that are not yet upgraded.
- You can still use the PDC to create new security principals and to replicate these changes to the Windows NT Server BDCs.
- Windows NT and Windows 9x workstations can use the PDC as a possible logon server.
- If the Windows 2000 Server PDC goes offline, or otherwise becomes unavailable, and no other Windows 2000 DCs exist in the domain, then a Windows NT BDC can be promoted to PDC.

The Effect of Upgrade on Access Control

Having moved the security principals into the Active Directory as part of the PDC upgrade process, one key concern is the effect this has on access to resources.

Security Identifiers (SIDs)

The Windows NT security model (the basis for Windows NT and Windows 2000 security) identifies account objects such as users, groups, machines, and domain trusts by SIDs. SIDs are domain-unique values, built when the user or group is created, or when the machine or trust is registered with the domain.

The components of a SID follow a hierarchical convention: a SID contains parts that identify the revision number, the authority—such as the domain—that issued the SID, and a variable number of sub-authority or Relative Identifier (RID) values that uniquely identify the security principal relative to the issuing authority.

Though there are well-known SIDs that identify generic groups and users across all systems, the majority of security principals we are concerned with are identified in the context of a domain, and thus they cannot be

moved between domains without their SIDs changing.

Authentication and Access Tokens

An essential component of the Windows NT security model is authentication, where a user is identified to the domain by presentation of credentials, usually in the form of a username and password. Assuming these credentials are acceptable, the system creates an access token for the user containing the SID of the user (the primary SID), and the SIDs of all the domain groups the user is a member of. Every process the user creates, for example by running an application, carries the user's access token.

This access token is used by the system to determine whether the user should be granted access to system resources.

Authorization and Security Descriptors

The counterpart of the user's access token is the security descriptor attached to resources such as files or printers. A security descriptor contains an Access Control List (ACL), which consists of a list of Access Control Entries (ACEs) Each ACE consists of a SID, together with an indicator that the security principal identified by the SID is granted or denied some sort of access to the resource.

From the above description, it can be seen that the affect of upgrade on SIDs is of crucial importance: if SIDs are altered in any way during upgrade, then it would follow that resource access would be affected.

In an upgrade, security principals remain in the same domain they were created in, and so the SIDs identifying them remain unchanged. As a result, resource access is unaffected by upgrade.

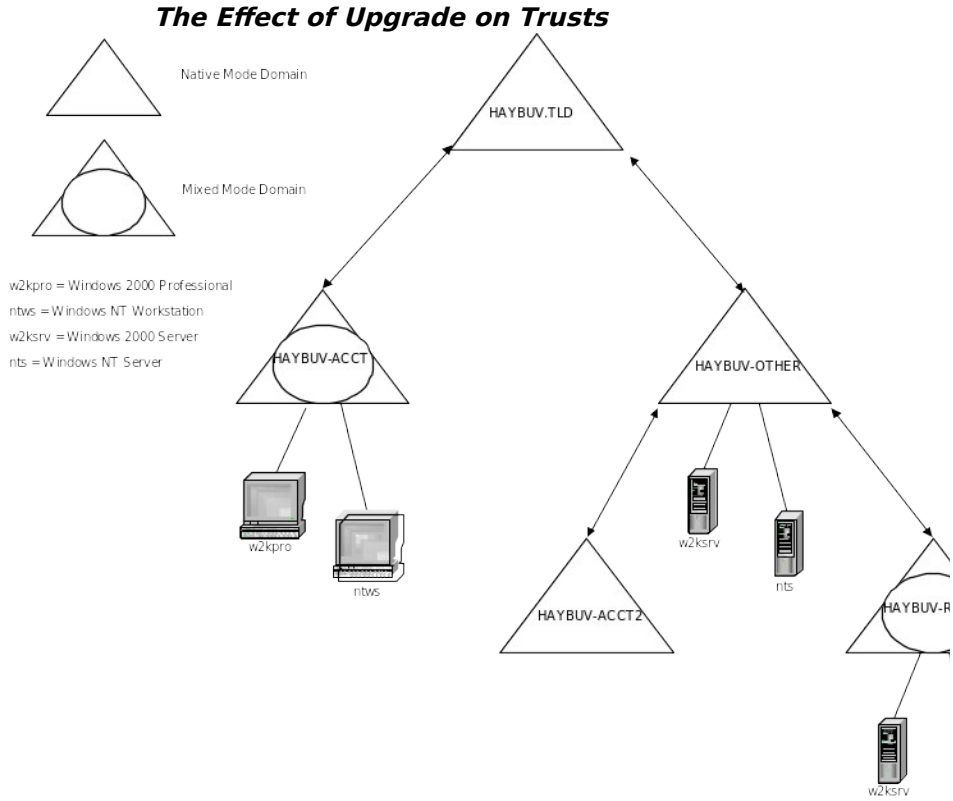


Figure 1. The

HAYBUV.TLD domain example

DCPROMO also installs the Kerberos software. Once this is complete, the authentication service and the ticket-granting service are started. If the administrator opted to join an existing tree, a two-way transitive Kerberos trust relationship is established to the parent domain. Any trust relationships created before the PDC was upgraded will still exist, but these will take the form of explicit one-way Windows NT style trusts.

Eventually, the DC from the parent domain copies all schema and configuration information to the new child domain. Once this information has been replicated, the upgraded domain is a fully functional member of the Active Directory domain tree, though until the administrator decides to switch the domain into Windows 2000 native mode it remains in mixed mode.

Active Directory-aware clients such as workstations running Windows 2000 Professional, or Windows 9x and Windows NT Workstation (running Active Directory client software) can now make use of Active Directory and perform actions such as querying global catalogs to locate resources and people. Clients will also be able to use the transitive trust relationships that exist within the forest to access resources throughout that forest. The manner in which this is achieved will depend on whether the client is running Windows 2000 or earlier operating systems such as

Windows NT or Windows 9x, and the upgrade state of the target domain.

Resources will be accessible across the forest via transitive trusts where the resources reside in:

- Native mode domains
- Mixed mode domains where all the DCs have been upgraded
- Mixed mode domains where the DC servicing the Kerberos or NTLM authentication request has been upgraded

In all other cases, clients will only have access to those resources that are available through existing Windows NT style one-way explicit trusts, which remain unchanged as a result of the upgrade.

We can use a couple of examples describing how NTLM and Kerberos authentication work to make the discussion clearer. Figure 1 (above) illustrates the scenario.

Using NTLM Authentication

Let us imagine a user logging onto the domain *haybuv-acct.haybuv.tld*, which is a mixed mode domain, from the Windows NT Workstation *ntws*, which is in the same domain. The user then tries to make a network connection to a Windows NT Server machine, *nts*, in the domain *haybuv-other.haybuv.tld*, which is a native mode Windows 2000 domain. Because *ntws* is a Windows NT client it will use the NTLM protocol.

Nts will determine that the domain name specified in the credentials passed to it, *haybuv-acct.haybuv.tld*, does not refer to its own account database, so it sends the logon request to a DC in its own domain for authentication. The DC checks the domain name, and because it doesn't match the DC's domain name, the DC checks to see whether the domain is a trusted one. The domains *haybuv-acct.haybuv.tld* and *haybuv-other.haybuv.tld* are both children of the same root— *haybuv.tld*—so transitive trust exists between the two domains, and the DC passes the logon request through to a DC in the trusted domain. That DC authenticates the username and password against its domain account database, and assuming the credentials match, passes the account-identification information back to the DC that contacted it, which in turn sends it back to the server.

Next, the server creates an impersonation token for the user, containing the user's SID and the SIDs of any domain groups the user is a member of. It then creates an impersonation thread in the user's security context, bearing the impersonation token, and attempts to access the resource on the user's behalf.

From this we can see that a down-level client in a mixed mode domain can access a down-level server in another domain in the tree using NTLM,

as long as the DC in the target domain is running Windows 2000 and thus understands transitive trusts. Because all trees in the same forest are linked by transitive trusts, the same would be true even if the two domains were in different trees.

By extension, it follows that if we were trying to access a resource on the Windows NT Server machine *nts* in the mixed mode domain *haybuv-res1.haybuv-other.haybuv.tld*, as long as the DC that the server passed the logon request to was running Windows 2000, the resource would be accessible via transitive trust across the forest.

Using Kerberos Authentication

Now let us imagine a user logging onto the domain *haybuv-acct.haybuv.tld* as before, but this time from the machine *w2kpro* in the same domain, which is running Windows 2000. The user wants to make a network connection to a Windows 2000 Server machine, *w2ksrv*, in the *haybuv-other.haybuv.tld* domain. Because *w2kpro* is a Windows 2000 client, it will attempt to use the Kerberos protocol.

Kerberos is a ticket-based protocol, where users are issued Ticket Granting Tickets (TGTs) by the ticket-granting service on the Key Distribution Center (KDC) running on the Windows 2000 DC, at initial logon to the domain. TGTs contain authentication information about the user encrypted with the domain master key, which can then be presented back to the DC as part of requests for additional session tickets to connect to other servers in the domain. Once the user has been granted a TGT, subsequent checks are quick and efficient, since the DC merely needs to decrypt the TGT to check the user's credentials. Session tickets are similar to TGTs, but are encrypted using a key shared between the server and the DC.

The Kerberos protocol, like NTLM, can operate across domain boundaries. A client in one domain can authenticate to a server in another if the two domains have established a trust relationship. When domains establish trust, they exchange inter-domain keys. Each domain's authentication service uses its inter-domain key to encrypt tickets to the other domain's ticket-granting service.

When a client wants access to a server in a remote domain, it asks the DC in its home domain for a referral ticket, a TGT that it can present to the ticket-granting service at the remote domain's DC. The local DC replies by sending the client a TGT encrypted with the remote DC's inter-domain key.

The client presents the referral TGT to the remote DC's ticket-granting service, asking for a ticket to a server in its domain. The remote DC decrypts the client's TGT with its inter-domain key. If decryption is

successful, the remote DC can be certain that a trusted authority issued the TGT, and so it grants the client a ticket to the requested server.

In Figure 1, as *haybuv-acct.haybuv.tld* and *haybuv-other.haybuv.tld* are children of the same root, and transitive trust exists between the two domains, a trust path can be built between the domains. On receiving the referral TGT, the DC in the target domain will check to see if it has a shared key for the server in question, and if it does it will issue a session ticket to the client. As *w2ksrv* is a Windows 2000 machine, a shared key will exist, so a ticket can be issued to *w2kpro*.

We can see from this that the important factors in this scenario are the existence of a DC in the target domain running the Kerberos ticket-granting service, and the existence of a shared key between the DC and the server. Windows 2000 DCs have the Kerberos service installed as part of the Active Directory installation process, and adding a member server to a Windows 2000 domain involves the creation of a shared key. From this it would follow that *w2kpro* would be able to access *w2ksrv.haybuv-res1.haybuv-other.haybuv.tld* using Kerberos as long as there was a Windows 2000 DC available to issue the session ticket.

If *w2kpro* was attempting to access a resource on a Windows NT machine such as *nts.haybuv-res1.haybuv-other.haybuv.tld*, Kerberos authentication would fail, and the client would fall back to attempting NTLM authentication as described in the previous section.

Upgrade and Resource Domains

A resource domain is a special type of domain, created to hold the accounts of resources such as servers and workstations. Resource domains were typically created in Windows NT to address two main situations:

- **SAM size limitation** - In Windows NT the maximum size recommended for the SAM account database is 40 megabytes (MBs). In an organization with many workstations and servers deployed, and many security groups, this might equate to much less than the 40000 accounts figure sometimes quoted. In order to scale an organization beyond this number it is recommended that user and machine accounts are stored in separate domains, termed an *account* or *master domain* for the former, and a *resource domain* for the latter. One would normally see resource domains created as second tier domains, with one-way trusts—a trusting relationship–—to either a single account domain (this topology is called the *master domain model*), or a number of account domains (the *multi-master domain model*).
- **Provide local administrative capability** - In a decentralized organization with geographically disparate facilities, it is often desirable to have local personnel authorized to administer resources.

In order to devolve this kind of responsibility in Windows NT it is recommended that resource domains are created with their own administrative structure. As in the above case, this would result in master or multi-master domain structures with one-way trusts to the account domains in the organization.

The one-way nature of these trusts ensures that the resource domain administrators, by default, only have administrative scope over the resource domain.

An important consideration when we consider the effects of upgrade on a resource domain is which of the two rationales described was foremost in the designer's mind when the domain structure was determined. If the domain was created to address the latter situation, you will want to consider the implications of upgrading a resource domain on your administrative model, because simply upgrading the resource domain and joining a forest will result in the creation of a two-way transitive trust between the child resource domain and the parent domain.

If the local administrators are less skilled, or you do not intend to grant them wider administrative rights in the Windows 2000 forest, you will probably want to consider the following options:

- **Upgrading the domain within the forest and using Windows 2000 delegation features -** Check the administrative groups in the resource domain and remove those who are not administrators in the master domains. If you only have local resource domain administrators, add one or more of your master domain administrators. They will be able to administer the domain while it is being upgraded. As a further precaution you should ensure that the resource domain administrators do not have administrative access to the DCs through local machine accounts.

  Once the PDC has been upgraded, create a new domain local group to hold your resource administrators, and add the users you removed from the administrative groups, as well as the modified administrative groups.

  Once the PDC has been upgraded, use Windows 2000 delegated administration to delegate appropriate authority to the newly created resource administrators' group.
- **Upgrading the domain in a new forest** - You could choose to upgrade your resource domain as a new tree in a new forest and maintain the link to the account domain by way of the pre-existing explicit trust to the account domain, which like a Windows NT explicit trust is unidirectional and not transitive. This would effectively mirror the pre-upgrade structure
- **Restructure into OUs** - Another alternative might be to rethink your domain structure and consider merging your resource domains into the upgraded master domain as OUs at a later time. This latter option

would obviously influence your thinking on the order of domain upgrade.

Upgrade and Administration

Once the PDC is upgraded to Windows 2000 Server and Active Directory is installed, the administrator can start using the new tools that come with Windows 2000 Server to create new objects that exist only in Active Directory, such as organizational units (OUs).

If you create OUs, you can begin organizing your domain by placing objects into them. This structure is visible only to computers with Active Directory client software. Without it, OUs do not appear; the domain appears unchanged. The PDC still exposes objects as a flat store to down-level clients.

An administrator working at a client without Active Directory can use only older administration tools.

Mixed Mode and Native Mode

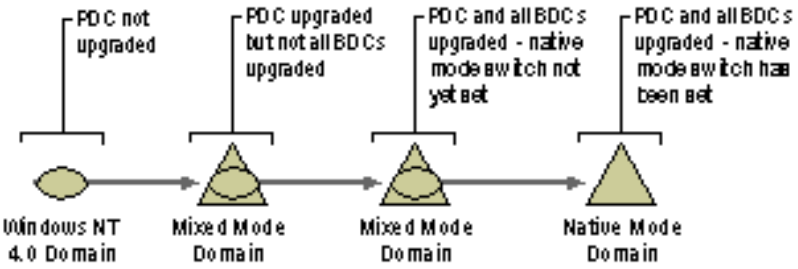

PDC not upgraded
PDC upgraded but not all BDCs upgraded
PDC and all BDCs upgraded - native mode switch not yet set
PDC and all BDCs upgraded - native mode switch has been set
Windows NT 4.0 Domain
Mixed Mode Domain
Mixed Mode Domain
Native Mode Domain

Figure 2. Stages of Upgrade

Mixed Mode

From the moment the PDC is upgraded to the time the administrator decides to switch the domain into Windows 2000 native mode, the domain is in mixed mode. Mixed mode provides maximum backwards compatibility with earlier versions of the operating system. It should be pointed out that mixed mode only relates to the authentication infrastructure of a domain, in other words, the domain controllers. Even when a domain has only Windows 2000 DCs, and has been switched to native mode, clients and servers running earlier versions of Windows NT, and clients running Windows 9x, can exist within the domain. This environment is known as a *mixed environment*.

You can leave the domain operating in mixed mode indefinitely, even if you have upgraded all the BDCs in the domain as well as the PDC.

Table 3 (below) summarizes Windows 2000 features available in mixed mode, and those only available by switching to native mode. If you are hesitant about switching to native mode you should consult your migration goals to determine whether staying in mixed mode

compromises them, and whether there are acceptable trade-offs.

Generally speaking, the only reasons to remain in mixed mode are:

- **Cannot upgrade BDCs** - In this scenario you have BDCs that cannot for some reason be upgraded or demoted to member servers. This might be the case where you have applications that must run on a BDC, but which for some reason will not run on Windows 2000.
- **Inadequate physical security of BDCs -** In any discussion of domain planning, security is an important consideration. A fundamental aspect of security is the physical security of the machine itself: any machine that is physically easy to access is vulnerable to attack. A consideration here could be the difference between single master updating of the SAM by the PDC alone, and Active Directory multi-master updating of the account database by all DCs.
  Because of the single-master nature of Windows NT directory updates, you might be comfortable with comparatively relaxed security on your BDCs. If this is the case, you need to reconsider this when upgrading them to Windows 2000 DCs. If you cannot upgrade security of your BDC appropriately you might consider demoting the BDC to a member server during upgrade, adding a new Windows 2000 DC in a different location, or possibly even reconsidering your proposed domain structure.
- **Fallback to Windows NT remains necessary -** One of the features of mixed mode, as will have become apparent, is the degree of backward compatibility. Mixed mode has the benefit of allowing new Windows NT BDCs to be added to the domain if a problem arises. In this situation, once the BDC has joined the domain, you could force a resynchronization of the account database. Then, as long as there are no Windows 2000 DCs in the domain, you could promote the BDC to PDC. You should plan for fallback or recovery, but at some point you will want to switch over completely to the new environment.

Native Mode

Once all DCs have been upgraded you can switch the domain from mixed mode to native mode. Several things happen during the switch:

- The domain uses only Active Directory multi-master replication between DCs, so support for Netlogon replication ceases.
- Because Netlogon replication is now switched off, you can no longer add new Windows NT BDCs to the domain.
- Because multi-master replication is enabled, the former PDC is no longer the master of the domain, and all DCs can now perform directory updates. One point that sometimes causes confusion is the continued existence of the PDC emulator role. Despite the multi-master nature of Active Directory, Windows 2000 has a number of

roles designated as Flexible Single-Master Operations (FSMOs), and PDC Emulator is a FSMO role. Usually, the former PDC will continue as PDC Emulator FSMO holder, which in a native mode environment means that password changes are replicated to it preferentially by other DCs.

- Windows 2000 group types such as universal and domain local groups, and group nesting, are enabled.

This switch to native mode is not performed automatically by running DCPROMO; the change must be initiated by the administrator, via the administrative interface.

Though you can get a great deal of benefit from upgrading your PDC and BDCs and remaining in mixed mode, certain attractive features such as universal groups, or group nesting, are not available in mixed mode.

The switch to native mode should be made as soon as possible.

Switching to Native Mode

Changing the domain mode from mixed to native mode is easy to do, but impossible to undo, so you need to bear in mind all of the factors discussed above to determine when to make the change. Do not change domain mode if you have or will have any Windows NT DCs.

To change the domain mode

1. Open the Active Directory Domains and Trusts snap-in, and right-click on the domain you want to administer in the left-hand tree view pane.
2. A context menu will appear. Choose **Properties** to show a tabbed dialog box, as shown in Figure 3.

3. On the **General** tab, click **Change Mode**.

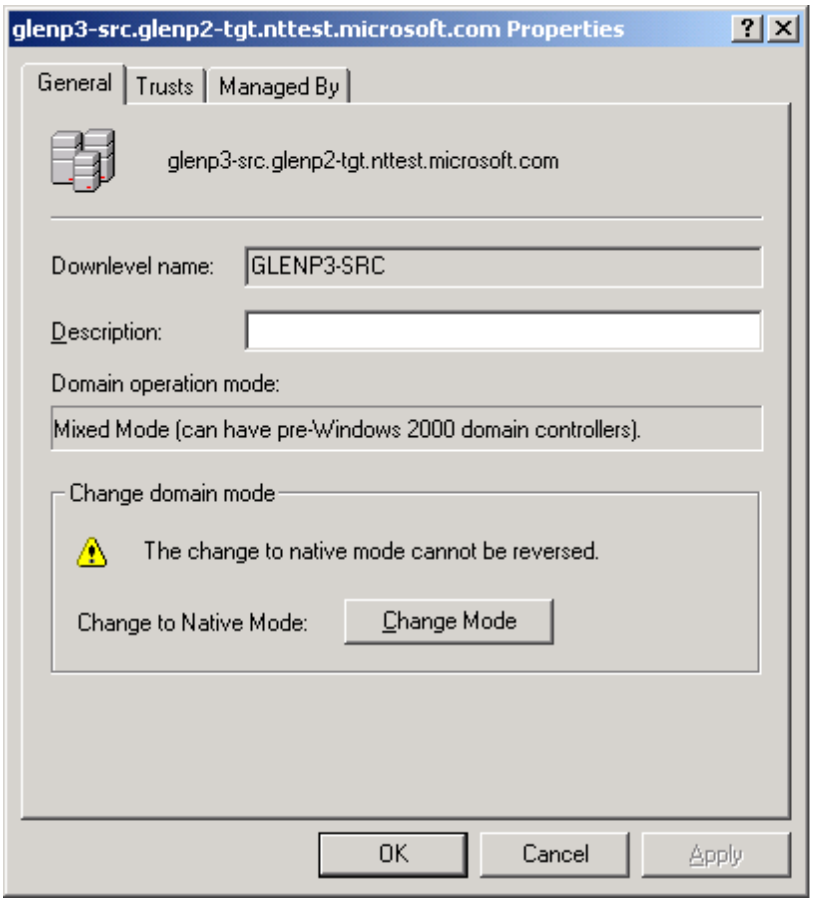


Figure 3. Change domain mode dialog

Table 3. Availability of Windows 2000 Features in Mixed Mode

Goal	Feature	Available in Mixed Mode?
Better manageability	Kerberos transitive trusts	Yes, but see the section "The Effect of Upgrade on Trusts" (above).
	Active Directory	Yes
	Active Directory organization units (OUs)	Yes, but only visible using Windows 2000 administration tools. Cannot be administered from Windows NT BDCs or member servers.
	New Active Directory	No, only Global and Local

	security groups	groups available.
	Windows installer	Yes
Greater scalability		
	Active Directory scalability	Yes, but only once all BDCs have been upgraded and are running the Active Directory. You should be wary of taking advantage of this feature as new Windows NT BDCs can still be added while the domain is in mixed mode. This is a feature that might be an important part of your fallback planning and so should not be compromised.
	Kerberos authentication	Yes, for Windows 2000 machines.
	Microsoft Management Console (MMC)	Yes
Improved security	Group policy	Yes, for DCs and other Windows 2000 machines in the domain.
	Security Configuration Manager (SCM)	Yes.
Improved availability	Active Directory multi-master replication	Yes, between PDC and BDCs that have been upgraded.

Windows 2000 Groups

In the previous section we mentioned new and richer security group functionality in Windows 2000.

Windows 2000 supports four types of security groups:

- Local
- Domain local
- Global
- Universal

Local Groups

Local groups have always existed in Windows NT. In terms of membership, they can have members from anywhere in the forest, other trusted forests, even trusted down-level domains. In terms of resource permission, though, they only have machine-wide scope, in that they can only be used to grant permissions on the machine on which they exist.

A special case for local groups in earlier versions of Windows NT is related to the local groups created on a PDC, which by virtue of replication of the domain SAM amongst BDCs, resulted in these groups being shared between the PDC and BDCs. In mixed mode, local groups behave in the same way on Windows NT and Windows 2000; in native mode, local groups on a DC become domain local groups (described in the next section).

Typically, local groups are used to grant ad-hoc access to resources on a local machine.

Domain Local Groups

Domain local groups are a new feature of Windows 2000, though similar in concept and use to the local groups created on the PDC in a Windows NT 4.0 domain (described above).

Domain local groups are only available in native mode domains. In terms of membership, they can have members from anywhere in the forest, other trusted forests, even trusted down-level domains. Domain local groups have only domain-wide scope in terms of resource permission, so they can only be used to grant permissions within the domain in which they exist. They differ from local groups in that they can be used on any Windows NT machine within the domain in which they were created.

Typically, domain local groups are used to gather together security principals from across the forest to control access to resources in the domain.

Global Groups

Windows 2000 global groups are effectively the same as Windows NT global groups. In terms of membership, they have domain-wide scope, but can be granted permissions in any domain, even in other forests and down-level domains as long as a trust relationship exists.

Universal Groups

Universal groups have forest-wide scope, that is, they can contain members from any Windows 2000 domain in the forest, and can be granted permissions in any domain, even in other forests as long as a trust relationship exists.

Though universal groups can have members from mixed mode domains in the same forest, members from such domains will not have the universal group added to their access token because universal groups are not available in mixed mode.

Though you can add users to a universal group, it is recommended that

you restrict membership to global groups.

Universal groups are only available in native mode domains.

Use of Universal Groups

Universal groups have a number of important uses. You can use universal groups to build groups that perform a common function within an enterprise: an example might be virtual teams. The membership of such teams in a large company would probably be nationwide or even worldwide, and almost certainly forest-wide, with the team resources being similarly distributed. Universal groups could be used as a container in these circumstances to hold global groups from each subsidiary or department, with the team resources being protected by a single Access Control Entry (ACE) for the universal group.

In using universal groups, an important factor to consider is that while global and domain local groups are listed in the Global Catalog (GC), their members are not, whereas universal groups *and* their members are listed, a fact which has implications for GC replication traffic. Care should be exercised in the use of universal groups. As a guide, if your entire network has high-speed connectivity you can simply use universal groups for all of your groups, and benefit from not having to bother about managing global groups and domain local groups. If, however, your network spans wide area networks (WANs), you can improve performance in several ways by using global groups and domain local groups.

If you use global groups and domain local groups, you can also designate any widely used groups that are seldom changed as universal groups.

Universal Groups and Access Tokens

In the discussion of universal group membership we touched on the fact that they can contain members from mixed mode domains, but that such members will not have the universal group’s SID in their access token. This is a consequence of the way access tokens are created in Windows 2000.

When a user logs on to a Windows 2000 native mode domain and has been authenticated, the Local Security Authority (LSA) on the DC where the user was authenticated retrieves the user’s global group memberships. The LSA then passes this information down to the workstation, where it is used to build the user’s access token. At the same time the LSA queries the GC for the user’s universal group memberships, which it also passes to the workstation.

If a user is a member of a universal group, the SID of that group is included in the access token on the workstation, and is added to the

authorization data in the TGT issued by the KDC.

Universal groups are not added to access tokens at any other time--for example, when impersonation tokens are created at member servers. As a consequence, if the universal group SID is not available when the user logs on—for example, where the user is logging on to a mixed mode domain—it will not be added subsequently.

Nesting groups

It is recommended that you do not create groups with more than 5,000 members. This is a guideline based on the fact that updates to the Active Directory store have to be capable of being made in a single transaction. Because group memberships are stored in a single multi-value attribute, a change to the membership would result in the whole attribute—in other words, the whole membership list—having to be updated in a single transaction. Microsoft has tested and supports group memberships of up to 5,000 members.

You can get around this limitation by nesting groups to increase the effective number of members. A further consequence is that you also reduce the replication traffic caused by replication of group membership changes.

Your nesting options depend on whether the domain is in native mode or mixed mode. The following list describes what can be contained in a group that exists in a native mode domain. These rules are determined by the scope of the group.

- Universal groups can contain user accounts, computer accounts, other universal groups, and global groups from any domain.
- Global groups can contain user accounts from the same domain and other global groups from the same domain.
- Domain local groups can contain user accounts, universal groups, and global groups from any domain. They can also contain other domain local groups from within the same domain.

Security groups in a mixed mode domain can contain only the following:

- Local groups can contain global groups and user accounts from trusted domains.
- Global groups can contain only user accounts.

Effects of Upgrade on Groups

Upgrading a PDC to Windows 2000 has no immediate affect on groups: Windows NT local groups become Windows 2000 local groups and Windows NT global groups become Windows 2000 global groups. The real change occurs when the domain is switched to native mode, at which point local groups on the PDC become domain local groups.

As stated earlier, when a user has been authenticated, an access token is created for the user containing their primary SID, together with the SIDs of any groups they belong to. We have noted how local groups on the PDC are a special case in Windows NT. This is reflected in how the access token of a user who is a member of such a group looks when created on the user's workstation after interactive logon, or at a DC for the purpose of impersonating that user to grant access to resources. Because the PDC local groups have no scope beyond the domain's PDC and BDCs, the user's access token on the workstation does not contain the SID for that group. When the user attempts to access resources on a DC, however, the impersonation token created for the user at the DC does contain the SIDs of any of the PDC local groups the user is a member of.

When the domain is switched to native mode, because domain local groups have domain-wide scope, the SIDs of any domain local groups of which the user is a member are added to the user's access token, even when the user is logging on interactively at a workstation.

Table 4. Groups and Domain Mode

Group Type	Membership	Scope	Available in Mixed Mode?
Local	Members from: • Same forest • Other trusted forests • Trusted down-level domains	Machine wide	Yes
Global	Members from the local domain	Any trusted domain	Yes
Domain local	Members from: • Same forest • Other trusted forests • Trusted down-level domains	The local domain	No
Universal	Members from the same forest	Any trusted native mode domain	No

NetBIOS and Windows Internet Name Service (WINS) in Windows 2000

NetBIOS is a high-level network-programming interface used in Microsoft networking components from the late 1980s onwards. Network resources are identified in the NetBIOS namespace by unique NetBIOS names. WINS is a service supplied as part of Windows NT Server to support registration

of dynamic mappings of NetBIOS names to IP addresses, and provide NetBIOS name resolution.

With the release of Windows 2000, support for the NetBIOS naming interface is no longer required for networking computers, and this coupled with the tight coupling of the Active Directory and Domain Naming System (DNS) will see the use of NetBIOS decline over time. In the meantime, upgrading your domain to Windows 2000 does not dispense with the need for NetBIOS support on your network, or affect the degree of support you currently have.

After upgrade you can look at discontinuing the use of NetBIOS and WINS if the following conditions are met:

- There are no clients such as Windows for Workgroups, Windows 9x, or Windows NT 4.0, and no servers running Windows NT 4.0 that use NetBIOS. NetBIOS names can still be required on your network to provide basic file and print services and support for many legacy applications used on client computers running under earlier versions of Windows operating systems.
  You should assess the impact of legacy applications and services in your testing plan.
- You have a pure Windows 2000 network and are sure that all computers and applications on your network are able to function using another naming service such as DNS. Network naming is a vital service for locating computers and resources throughout your network, even where NetBIOS names are not required.

The Windows 2000 WINS client caches resolved names locally, and uses a component called the Caching Resolver to look in this cache before submitting a query to DNS. The host file is cached as soon as the client starts and any updates to the host file are reflected in the cache immediately. The name resolution sequence is as follows:

1. Attempt name resolution from the client cache.
2. If resolution from the client cache fails, the client attempts name resolution through DNS.
3. If DNS name resolution fails, the client attempts resolution through WINS.

As a result, assuming that you have implemented sufficient change control over your new clients as you have upgraded them, and once all legacy conditions have been removed, the switch away from NetBIOS and WINS should be seamless.

Availability of LAN Manager Replication Service (LMRepl)

Windows NT Server provided a replication facility known as LAN Manager Replication (LMRepl). LMRepl is often used to replicate logon scripts from an exporting DC to a number of importing DCs in the domain. The File Replication Service (FRS) replaces this service in Windows 2000 Server.

Windows 2000 Server does not support LMRepl in mixed or native mode, so if you have been using LMRepl, you will need to include in your upgrade plan a strategy for using FRS to provide the same functionality.

The LMRepl Process

LMRepl uses the concept of import directories and export directories. An administrator can configure LMRepl by selecting a server on which to host an export directory and a number of servers to host import directories. The servers hosting the directories do not need to be DCs; they can be ordinary member servers.

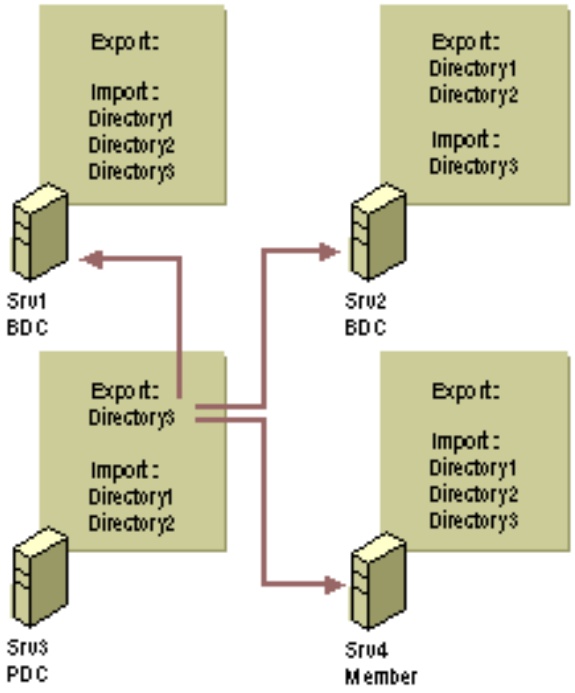


Figure 4. The LMRepl Process

The FRS Process

FRS (File Replication Service) in Windows 2000 Server is automatically configured so that every domain controller has a replicated System Volume (SYSVOL). Any change you make to a logon script stored in the SYSVOL of any domain controller is replicated in multi-master fashion to other domain controllers. Unlike LMRepl and the hosting of import and export directories, in FRS only domain controllers can host the SYSVOL.

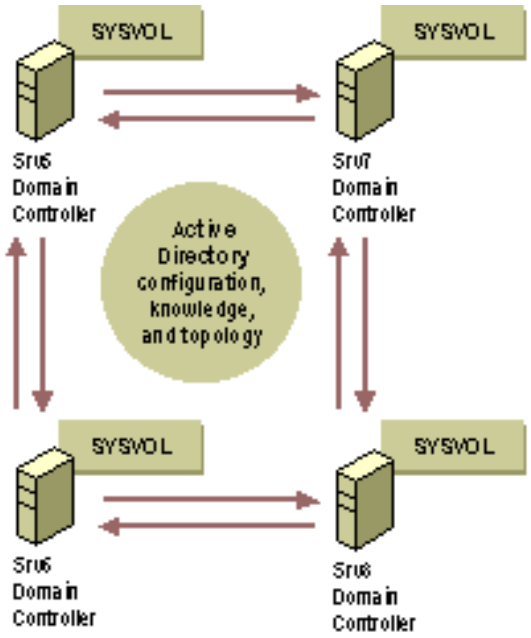


Figure 5. The FRS process

Maintaining LMRepl Services in a Mixed Environment

As we have seen, during an upgrade you can have a mixed environment of Windows NT 4.0 BDCs and servers operating with Windows 2000 domain controllers. Because Windows 2000 Server does not support LMRepl, maintaining LMRepl services in a mixed environment could be an issue. To provide this support you will need to create a bridge between LMRepl and FRS so both services can operate. You could do this by selecting a Windows 2000 DC whose SYSVOL would be used to populate the export directory of the Windows NT BDC hosting LMRepl. A sample script, which can be regularly scheduled to perform the necessary file copying, is available from Microsoft.

LMRepl and Upgrade

In order to maintain availability of LMRepl during upgrade, you should ensure that the server hosting the export directory is upgraded only after all the other servers hosting import directories have been upgraded. If the server hosting the export directory is the PDC, you should select a new export server and reconfigure LMRepl.

Using Remote Access Services (RAS) in a Mixed Environment

If you are using RAS to provide your users with remote access to the corporate network, you might want to consider upgrading your RAS server early on in the process of upgrading member servers. This is because Windows NT checks RAS properties such as availability of RAS access or dial-back for a user.

RAS is an example of a Windows NT *service*. A service is a type of background process, designed to run continuously with no user interface, usually fulfilling some kind of server function such as a Web server or an FTP server. Because the service must run even when no users are logged on to the system, it runs in the security context of a special service account, usually the system account, also known as *LocalSystem*.

The System account is a special account known only to the local machine, and is granted all privileges available on the operating system. When a service logs on as LocalSystem, it logs on with NULL credentials. In other words, it does not provide a username or password. This means that this account cannot be used to access network resources relying on NTLM authentication unless the remote machine allows access with NULL credentials (often referred to as a NULL session). In Windows NT, RAS uses the LocalSystem account.

By default, the Active Directory will not accept querying of object attributes via NULL sessions, so in a mixed environment a Windows NT RAS server will not be able to retrieve a user's RAS properties unless the following conditions are met:

- The domain is in mixed mode and the Windows NT RAS server is also a BDC. In this case RAS will have local access to the SAM.
- The domain is in mixed mode and the Windows NT RAS server contacts a Windows NT BDC, which would result in behavior identical to current Windows NT behavior. This might lead to some inconsistencies of behavior.
- The domain is in mixed mode or native mode and the Active Directory security has been loosened to grant the built-in personality *Everyone* permissions to read any property on any user object. The Active Directory installation wizard (DCPROMO) allows the user to select this configuration by means of a *weaken the permissions* option on certain Active Directory objects.

You should use this last workaround only after careful study of its impact on the security of Active Directory. If you feel that it conflicts with your security requirements you should adopt the recommended approach, which is to upgrade the Windows NT RAS server to Windows 2000 and make it a member of a Windows 2000 mixed or native domain. This would also have the benefit of avoiding inconsistent behavior while the domain is in mixed mode, as described in the second condition.

These factors should be borne in mind in your upgrade plan, an implication being that you should consider upgrading your RAS server first in the server upgrade process.

Supported Upgrade Paths

An important consideration in planning your upgrade to Windows 2000 will be the operating systems you have already deployed in your

enterprise, and whether it is possible to upgrade them directly to Windows 2000.

Table 5 (below) contains a list of supported upgrade pathsfor Windows 2000. In the cases where an upgrade is required, but a direct upgrade is not supported, you will upgrade via another operating system such as Windows 9x, Windows NT 3.51/4.0 for workstations, or Windows NT 3.51/4.0 for servers. This will need to be reflected in your upgrade plan.

Table 5.Supported Upgrade Paths

Platform	Upgrade to Windows 2000 Professional	Upgrade to Windows 2000 Server
Windows 3.x	No	No
Windows NT 3.1	No	No
Windows NT 3.1 Advanced Server	No	No
Windows NT 3.51 Workstation	Yes	No
Windows NT 3.51 Server	No	Yes
Windows 9x	Yes	No
Windows NT 4.0 Workstation	Yes	No
Windows NT 4.0 Server	No	Yes

Order for Upgrading Domains

In the discussion to date we have covered the order of upgrade for DCs in a domain. There is little room for discussion on this point: you must upgrade the PDC first, then the BDCs. The question of which domain to upgrade first is more problematic, and the answer may vary depending on your circumstances. For example, if you are planning to restructure certain domains out of existence, there might be little point in upgrading them first.

Though your situation may change this, a general recommendation is that you should consider the following order for upgrading your domains:

1. Account domains
2. Resource domains

Upgrading Account Domains

As a general rule, you will get the most benefit from upgrading your account domains earliest because in most cases there will be more users to administer than computers. By upgrading your account domains to

Windows 2000 you will benefit from:

- **Improved scalability of Active Directory** - Many organizations are pushing the upper bounds of the recommended SAM size with their existing numbers of users and groups.
- **Delegated administration** – The ability to delegate administrative capability at very fine granularity, without the necessity to grant absolute power.

Order of Account Domains

If you have more than one account domain, the following guidelines should help you choose in which order to upgrade them:

- **Maintain control** – Though you will have tested your upgrade strategy in a lab, or via a pilot, the first live migration will be the riskiest. To mitigate risk you should upgrade domains where you have easiest access to the DCs.
- **Mitigate risk and disruption** - If there is more than one domain to choose from in any situation, upgrade the smallest first so that you minimize disruption to the most possible users, particularly while you are gaining experience of the process.
- **Get the job done** – Once you have gained experience of and confidence in the process, move onto the bigger account domains.
- **Targets of account domain restructure** – If you are planning to restructure your domains, you should look to upgrade the likely targets of restructure early in the process. You cannot consolidate domains into a target that does not exist.

Order of Resource Domains

If you have more than one resource domain the following guidelines should help you choose which order to upgrade them:

- **Provide the features applications need** – First you should upgrade domains where you are deploying applications that demand Windows 2000 infrastructure or features, such as the Active Directory required by Microsoft Exchange Platinum (the next major release of Microsoft Exchange).
- **Domains with many workstations** - Next you should upgrade domains with many workstations, so that you can take advantage of Windows 2000 infrastructure such as Microsoft IntelliMirror™.
- **Targets for resource domain restructure** – Just as with account domains, if you are planning restructure of your domains, you should look to upgrade the likely targets of restructure fairly early on.

Upgrading Workstations and Servers

Though the focus of this paper is domain upgrade and consolidation, you should not take it that deployment of Windows 2000 workstations and member servers should be postponed until the domain infrastructure has been upgraded. Using Windows 2000 workstations and servers in an existing Windows NT environment will still have a number of benefits.

Domain Restructure

Domain upgrade is a process designed to maintain as much of your current environment as possible, including your domain structure. Domain restructure on the other hand is a process designed to allow you to redesign the forest according to the needs of your organization. Though restructure can result in any number of different outcomes, typically the result is some rationalization of the current structure, and perhaps a move to fewer larger domains.

In the past, there have been a number of third-party directory management tools that have provided domain-restructuring support for Windows NT. For the first time, Windows 2000 provides native functionality to enable domain-restructuring scenarios, namely:

- Security principals can be moved from one domain to another while maintaining pre-move access to resources.
- DCs can be moved from one domain to another without complete reinstallation of the operating system.

Microsoft is making available a graphical tool to make domain restructuring easier, together with some scriptable COM components and command line utilities to aid restructuring operations. These will be useful to many customers implementing the scenarios described below.

When Should You Restructure?

Restructuring is appropriate in three sets of circumstances:

- **Post-upgrade -** The most common time for organizations to perform domain restructuring will be as the second phase of migration to Windows 2000, following an earlier upgrade.
  The upgrade will have covered the ‘easier’ migration cases, such as groups of domains where the trust structure is essentially correct, and where there are no administrative implications.
  In these circumstances restructuring will probably be aimed at reworking other parts of the domain structure to reduce complexity, or to bring resource domains with untrusted administrators into the forest in a secure way.
- **Instead of upgrade -** Some organizations might feel that their current domain structure is for some reason unsalvageable, or that they cannot afford to jeopardize the stability of the current production environment during migration. In such cases the easiest migration path might be to design and build an ideal or pristine

Windows 2000 environment separate from the current production environment. Here the new Windows 2000 forest is intended, over time, to become the production environment.

Once the new forest has been built, restructuring will begin with a pilot, where a number of users, groups, and resources are migrated to the new environment to act as an advance party, ensuring that business can carry on as normal in the new structure.

On successful completion of this phase, the pilot will transition into a staged migration to the new environment. At some point Windows 2000 will become the production environment. The old domain structure will be decommissioned, and the remaining resources will be redeployed.

- **Post-migration -** In this case, restructuring takes place as part of general domain redesign in a completely Windows 2000 environment. This might occur several years down the line, when for some reason —such as organizational change or corporate acquisition—the current structure becomes inappropriate.

The main focus of this chapter is on the initial migration from Windows NT 4.0 to Windows 2000, namely the first two cases

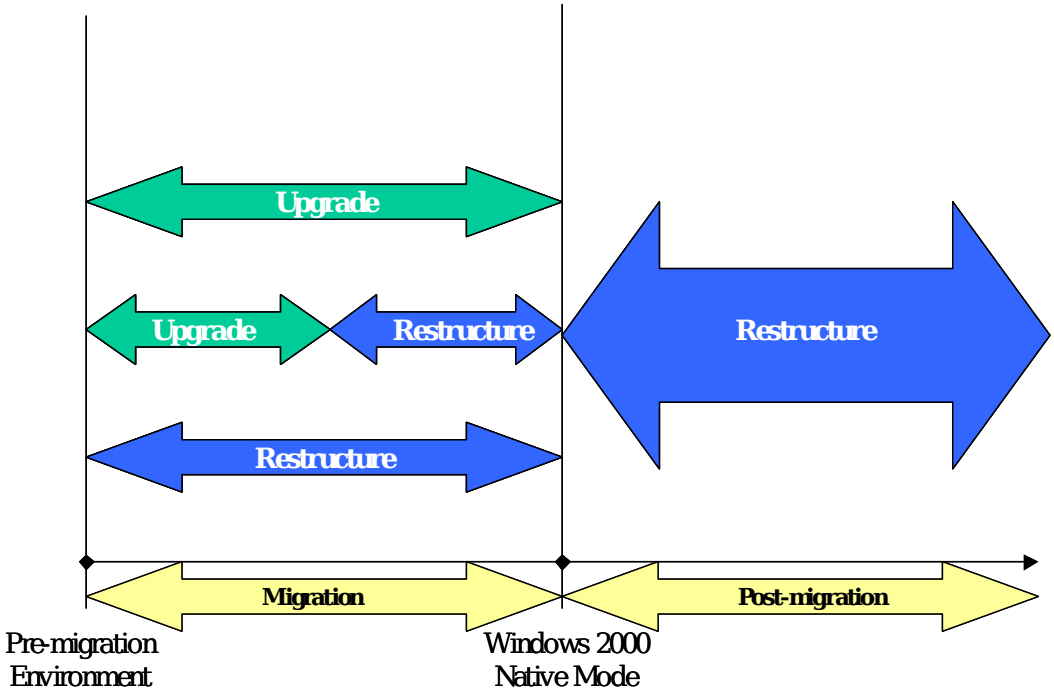


Figure 6. When restructure can occur

Why Would You Restructure?

You might wish to carry out domain restructuring for a number of reasons, but the major driver is likely to be to take advantage of Windows 2000 features that allow you to better structure your domains to reflect the requirements of your organization, such as:

- **Greater scalability -** You might have designed your previous Windows NT domain structure around the size limitations of the SAM accounts database, leading you to implement a master or multi-master domain model. With the vastly improved scalability of the Active Directory, scaling to literally millions of user accounts or groups, you could restructure your current Windows NT domains into fewer, larger Windows 2000 domains.
- **Finer granularity of administration -** In your current model you might have implemented resource domains to allow delegation of administrative responsibility. Windows 2000 OUs can contain any type of security principal, and administration can be delegated as appropriate. In many instances, converting resource domains into OUs will be a more appropriate method for delegating administration.
- **Simpler administration -** To allow delegation as described above, or perhaps as a result of corporate acquisition, your domain structure might be connected by a complex mesh of trusts. Some of these domains might be better implemented as OUs, thus simplifying administration. Alternately, you could redesign your domain tree and benefit from fewer, bi-directional transitive trusts.

The scenarios we will describe are not predicated on your having completed an initial upgrade.

Implications of Domain Restructure

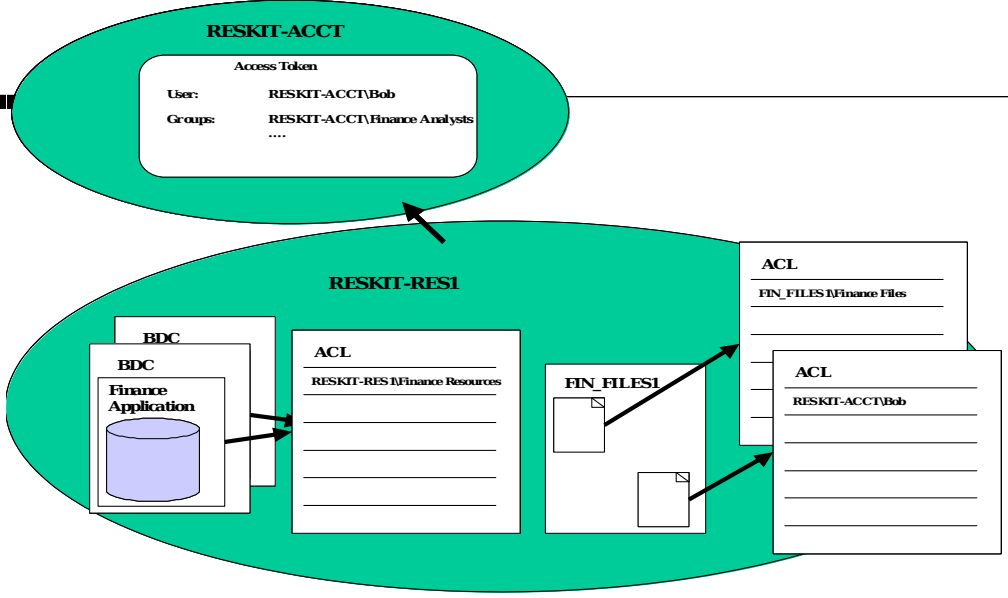
The fundamental enablers for restructuring scenarios are, as mentioned above, the ability to move security principals and DCs between domains. In this section, we will examine the implications of these features and the impact they have on planning your restructure.

The Effect of Moving Security Principals on SIDs

We talked earlier about the domain-relative nature of SIDs. A consequence of this is that when you move a security principal such as a user or a group between domains it has to be issued with a new SID for the new domain.

As we saw in the section on upgrade, access to resources in the Windows NT security model is carried out by the operating system looking at the user's access token and evaluating the user's primary SID, and the SIDs of any groups the user is a member of, against the ACL on the resource's security descriptor. ACLs are made up of lists of SIDs indicating granting or denial of access to the resource for the security principals identified by the SIDs. Thus, changing the SID has far-reaching implications.

To illustrate these implications, based on what we know of Windows NT security, we will use an example. Bob is an employee of Hay Buv Toys, and has an account in the Windows NT 4.0 master domain *HAYBUV-ACCT*. Bob is a member of the global group *Finance Analysts* in the same domain.



Hay Buv Toys uses a Win32-based financial application that runs on a number of Windows NT Server machines in a resource domain *HAYBUV-RES1*. In order to make use of the fact that local groups created on the PDC are shared amongst all DCs in the domain, the servers the application runs on are running as BDCs in the domain. A shared local group called *Financial Resources* has been created on the PDC, and is used on ACLs on the files used by the application. The global group *HAYBUV-ACCT\Finance Analysts* is a member of *HAYBUV-RES1\Financial Resources*.

Bob also has access to a file server, *FIN_FILES1,* in the resource domain. *FIN_FILES1* is simply a Windows NT Server machine running as a member server. *FIN_FILES1* uses a local group called *Finance Files* on the ACLs of files relating to Bob's group. *HAYBUV-ACCT\Finance Analysts* is a member of *FIN_FILES1\Finance Files*. Bob works on some private projects and has a directory on *FIN_FILES1* that is protected so that only he can access the files in that directory. This directory has an ACL that contains a single entry allowing Bob full control of the directory.

Figure 7. Resource access example

Moving Security Principals

As indicated earlier, moving a security principal between domains has the effect of changing the security principal's SID.

We can appreciate the implications of moving security principals by tracing what would happen if Bob's account were moved to another domain as part of a migration involving domain restructuring.

For the purposes of this example we will assume that *HAYBUV-ACCT* has been upgraded to Windows 2000, and has joined the corporation's new Windows 2000 forest as a child of the root domain *HAYBUV.TLD*. *HAYBUV-ACCT* has been switched to native mode, but it is intended to be restructured, with its members being moved to a new Windows 2000

domain *HAYBUV-ACCT2* elsewhere in the forest.

Affects of the Move on Bob's Global Group Membership

HAYBUV-ACCT *Bob* is a member of the global group *HAYBUV-ACCT*\ *Finance Analysts*. As a global group can only contain members from its own domain, Bob's move to the new domain would result in his new account not being able to be a part of this group, which would mean he would lose access to resources available to this group.

Assuming sufficient trust existed between the new domain and the resource domain, it would appear that we could fix this in a number of ways:

- **Adding Bob's new SID to resource ACLs** - We could maintain access to resources by adding Bob's new SID to the ACLs on all the resources he formerly had access to as a member of the group. This would be time-consuming and messy for a number of reasons:
  - Many domain-restructuring operations will be carried out incrementally over a period of time, and during this transitional period we could not guarantee that new resources would not be created for the *Finance Analysts* group. Because of this we would have to continue to re-permission during the migration periods while the two groups coexisted.

  - Microsoft recommends that groups, rather than individuals, be used in ACLs. This is because the memberships of groups carrying out specific job functions can change over time, and it is easier to take a user out of a group than it is to change the ACLs of numerous resources that reference the user. What would happen if Bob's job function changed and he no longer needed to be a member of the *Finance Analysts* group?

- **Moving the group** – Just as we can move security principals in Windows 2000 we could move the group itself to the new domain. Although, as the ACLs that reference the group also reference its SID, we would have to do a similar exercise to the first option; in other words, we would have to re-permission resources to refer to the new SID.
- **Creating a parallel group in the target domain** - If we were to move the group, a problem would occur if the migration plan did not envisage moving all the group members in one transaction. This would mean that the group would have to be maintained in the old domain, and a new parallel group created in the new domain. Resource access would be maintained for the original group and its members, but we would need to re-permission resources to grant access to the new group. Again, we would have to continue to re-

permission while the group existed in both domains.

Effects of the Move on ACLs Directly Referencing Bob

Bob is also granted access to some resources on the member server *FIN_FILES* directly. His SID appears on ACLs on that machine. It is perfectly legal to add users to ACLs on resources, but an affect of Bob's move would at first sight appear to be a requirement to re-permission any resources on that server, adding his new domain SID.

Microsoft has always advised the use of groups rather than individuals in resource ACLs., In many organizations, though, this might not have been enforced, resulting in a requirement to seek out and update all references to a moved user in all ACLs on resources across the organization.

SIDhistory

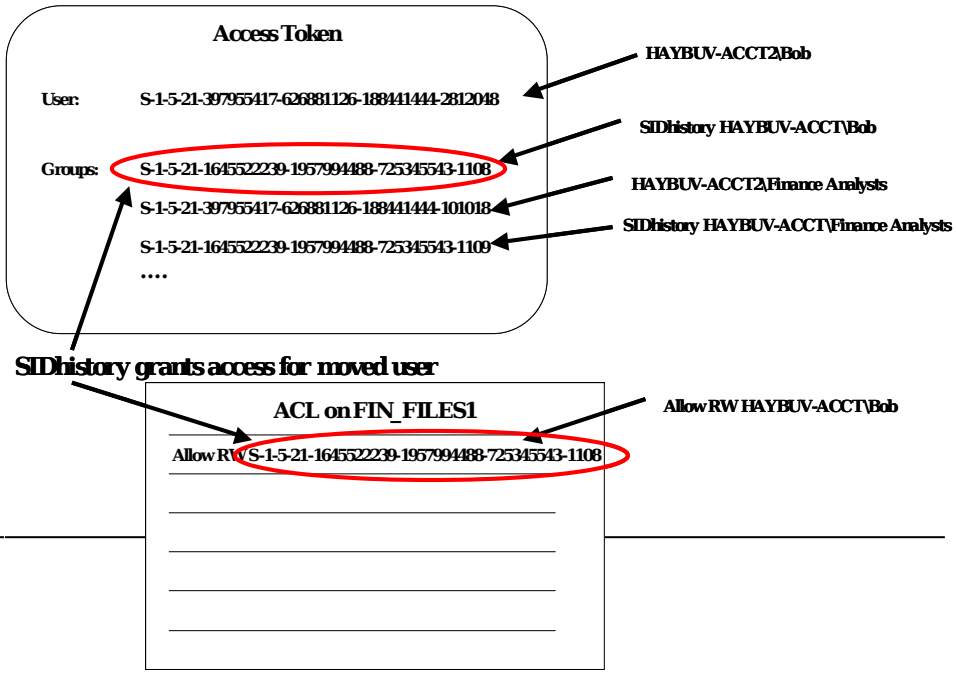
In many instances the activities just described are made unnecessary as a result of a new Windows 2000 directory attribute called SIDhistory.

SIDhistory is an attribute of Active Directory security principals, and is used to store the former SIDs of moved objects such as users and security groups.

When a user or group is moved using new Win32 APIs, or the basic utilities provided by Microsoft and built on top of them, the SIDhistory attribute of the object representing it in Active Directory is updated with its former SID as part of the operation. When the user then logs on to the system, as well as the user's group memberships, the system retrieves the entries in the user's SIDhistory and adds them to the user's access token.

Because groups can be moved, the system also retrieves the SIDhistory attributes of all the groups the user is a member of and adds these to the user's access token.

These SIDhistory entries in the token look to the system like normal group memberships during authorization checks, so they can grant



appropriate access even on down-level systems that know nothing about Windows 2000 or Active Directory.

Figure 8. Resource access granted through SIDhistory

Moving Users and Groups While Updating SIDhistory Using MoveTree.exe

MoveTree is the name of a domain migration basic utility available in Windows 2000 Beta 3.It enables the movement of directory objects, including security principals, between Windows 2000 domains in the same forest. At the same time, it updates the SIDhistory of users or groups moved. MoveTree will move objects such as OUs, groups, and user and computer accounts.

Constraints on the Use of MoveTree

There are some constraints on the use of MoveTree. The utility requires:

- **Mixed or native mode source domain** - The domain from which the object is being moved (source) must be a Windows 2000 mixed or native mode domain
- **Native mode target domain** - The domain to which the object is being moved (target) must be a Windows 2000 native mode domain
- **Source and target domains** – Should be in the same forest
- **Empty global groups** – Though MoveTree will move OUs and their contents, and with a native mode source domain, universal groups and their contents, MoveTree will only move empty global groups. Currently, using MoveTree requires that you depopulate global groups before moving them, and then repopulate them at the destination.
- **Closed sets -** Objects must be moved in *closed sets*. The meaning of the term depends on whether we are referring to the movement of users and global groups, or the movement of computers, which might use shared local groups in the case of Windows NT DCs, or domain local groups in the case of Windows 2000 DCs, servers, and workstations in a native mode domain.

Closed Sets and Moving Users and Global Groups

As we saw earlier, a global group can only contain members from its own domain. To maintain access to resources protected by ACLs referring to global groups, when a user moves between domains, any global groups the user is a member of must also be moved. A corollary of this is that if a global group is moved, its members must also be moved.

In this instance a closed set of users and global groups is a set where for each user being moved all of its global groups are also being moved, and for each group being moved all of its members are also being moved.

If the source domain is a native mode domain, global groups can also contain other global groups, in which case for each nested group, all of its members and all global groups its members are members of must be moved.

Closed Sets and Moving Computers

Shared local groups and domain local groups only have scope within the domain in which they were created. If such a group moved, any references to the group in ACLs in the source domain would be irresolvable.

In this instance a closed set of computers and shared or domain local groups is a set where for each such group being moved, all computers in the domain containing ACLs referencing the group are also moved. Also, for each computer being moved, all shared or domain local groups referenced in ACLs on its resources are also being moved.

Ways Around Closed Sets

The restrictions on the movement of populated global groups and closed sets are particularly onerous. Depopulating and repopulating large global groups can be time consuming, and in some cases the smallest closed set achievable will be the whole source domain.

In such cases there are three possible approaches to solving the problem:

- **Parallel groups** - Creating parallel global groups in the target domain for each group to be moved, then locating all resources in the enterprise containing ACLs referencing the original group and re-permissioning them to include reference to the parallel group. In the case of moving global groups, where the group could be referenced on resources in any trusting domain, or domain local groups from native mode source domains where domain local groups can be used on any machine in the domain, this is likely to be a large undertaking.
- **Universal groups** - Switching the source domain to native mode, then changing the group type of the groups to be moved to universal. Because universal groups have scope across the entire forest, changing the groups to universal groups means that they can safely be moved while still maintaining access to resources left behind.
  Some care should be exercised when using this approach, since as we discussed earlier the membership of universal groups is

stored in the GC, which has implications for GC replication traffic. Because of this you might want to use this approach as a transitional strategy while users and groups are being migrated to the new domain. Then, once the migration has been completed, change the groups back to their original types.

- **Cloning** - Cloning groups from the source domain into the target domain, while retaining SIDhistory. This technique has some constraints of its own, and is dealt with in the section “Cloning Security Principals”.

Cloning Security Principals

Up until now we have thought of restructure as being enabled by moving security principals. In some ways moving security principals can be regarded as a destructive operation, because in moving the security principal we are cutting off fallback to the old account in the case of problems during the migration.

Many customers have indicated a desire to do an incremental migration of users to a new Windows 2000 domain, while maintaining users’ old accounts in the source domain, for the purposes of disaster recovery during pilot or mass migration. We refer to this functionality as *cloning*, a feature Microsoft has enabled for Windows 2000 Beta 3. Microsoft is providing some basic utilities known as ClonePrincipal, which allow a user or group to be cloned from a Windows NT 4.0 or Windows 2000 source domain to a Windows 2000 native mode domain, without removing the source account. At the same time, the utility adds the original account SID to the SIDhistory of the new account to maintain resource access.

ClonePrincipal

ClonePrincipal consists of a scriptable component enabling the cloning of security principals together with usable sample scripts illustrating the usage of the component. The scripts will be customizable by administrators with knowledge of Microsoft Visual Basic® (VB).

Because cloning is a security sensitive operation, there are some constraints on the use of ClonePrincipal:

- The source and target domains cannot be in the same forest.
- The SID of the source account must not already exist in the target forest either as a primary SID or in the SIDhistory of any security principal.
- To execute ClonePrincipal the user must have Domain Administrator privileges in both the source and target domains.
- Auditing must be switched on in the target domain. It is also recommended that auditing be switched on in the source domain.

Establishing Trusts

In cloning scenarios, where the source and target domains are in separate forests, establishing the trusts necessary to guarantee resource

access will be one of the first steps.

Netdom.exe is a tool provided with the Windows 2000 Server Resource Kit to carry out such tasks as enumerating domain trusts, and establishing new trusts.

The tool is also useful for creating machine accounts and updating the domain membership of a workstation or server machine.

Moving Servers

In our previous example, Bob has access to some resources on the member server *FIN_FILES1*, via ACLs referencing a machine local group *FIN_FILES1\Finance Files*, and referencing his domain account directly.

We have discussed the implications of moving DCs, namely the need to ensure that shared local groups and domain local groups are maintained, but what about the implications of moving a member server such as *FIN_FILES1* or a workstation?

Assuming that the server was moved to a domain with trust to Bob's new account domain, we have seen that SIDhistory would ensure that Bob could access resources with ACLs referencing him directly. ACLs referencing the machine local group would also continue to function by virtue of the fact that the group exists in the local machine's account database. Thus, it is unaffected by the move, and its SID would not need to be changed.

Windows NT 3.51 and SIDhistory

When planning domain restructure, you should be aware of a known problem related to the use of Windows NT 3.51 in Windows 2000 domains.

The problem concerns authentication and authorization, specifically, the way access tokens are constructed in Windows NT 3.51. When a user is authenticated, either interactively at a Windows NT 3.51 workstation, or over the network at a Windows NT 3.51 server, the token is built using only SIDs relative to the user's account domain, and local groups from the server or workstation where authentication is taking place. The result is that Windows NT 3.51 access tokens cannot contain universal groups from outside the account domain, or domain local groups from the resource domain. Since by definition entries from the user's SIDhistory, or the SIDhistory of any groups the user is a member of, will be from domains other than the account domain, these will also be excluded from the token.

The implication of this is that with Windows NT 3.51, SIDs from domains other than the account domain of the user logging on are ignored when evaluated for access control.

In most cases this results in access being denied, though this might not

be the desired result.

ClonePrincipal and Resolving SIDs

Another problem that should be borne in mind when formulating your domain restructure plan relates to the way SIDs are resolved to usernames. In some circumstances when a user tries to look at the security properties of a file or folder on a Windows NT File System (NTFS) partition, the SIDs of users and groups that have been cloned may not be resolved and will be labeled as *Unknown User*. This could cause management problems, in that administrators might be hesitant about editing ACLs for fear of preventing legitimate resource access.

To understand this problem it is useful to look at the way the API used to resolve SIDs attempts resolution. The API attempts resolution in the following order:

1. It attempts to find a name for the SID by first checking a list of well-known SIDs.
2. If the SID does not correspond to a well-known SID, the function checks built-in and administratively defined local accounts.
3. Next, the function checks the primary domain.
4. SIDs not recognized by the primary domain are checked against the trusted domains corresponding to their SID prefixes.
5. Next, the function will attempt to check the SID against the SIDs of all accounts in all domains in the Windows 2000 forest, including SIDs that appear only in the SIDhistory field of an account in the forest. To perform these look ups, the function needs to be able to query the GC of the forest.

This has a number of implications:

- To be able to resolve the name without querying the GC, the account that was cloned needs to be in a trusted domain, and needs to still exist.
- To resolve the name as described in step 5 above, however, the computer from which the function attempts the check must either be running Windows 2000, or be in a Windows 2000 native mode domain.

This problem will not occur where users have been moved using MoveTree. These moves are limited to the same Windows 2000 forest, so this will only occur where users have been cloned between forests.

To avoid this problem, you should consider:

- Migrating all workstations to the target forest as soon as possible after users have migrated.
- Re-ACL resources across the enterprise to remove references to old SIDs.
- Maintaining the source account in the old domain until resources

have been re-ACL'ed and all references to the old SIDs have been removed.

Profiles and SIDhistory

When formulating your domain restructure plan you should be aware of the implications of migrated users having new SIDs on profile use.

During user log on to a Windows NT workstation, the system will load a profile for that user, containing user-specific configuration information. The system locates this profile by picking up the user's SID, and looking in the registry under the key *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList* for a key named after the string representation of the user's SID.

An implication of this behavior is that a user logging onto their workstation after migration might lose access to their logon profile, since their primary SID will have changed and their old profile will be stored under their old primary SID.

Where a user has been moved between Windows 2000 domains in the same forest using a tool such as MoveTree, this will not be a problem since MoveTree preserves the user's GUID. A new feature of the profile handling code in the Windows 2000 client takes advantage of this invariance of the GUID, so that on failure to find a user's SID in the *ProfileList* key, the system will retrieve the user's GUID and look in the registry under the key

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileGuid

for a key named after the string representation of the user's GUID. If it finds such a key, it will search it for a value called the SidString, which provides a fallback mapping of the user's GUID to the user's SID. The system can then look under the *ProfileList* key for the SID contained in SidString, and if it finds it, rename the key to the string representation of the user's new SID. The system will in this case also change the SidString value.

From the above description, it can be seen that an issue with loss of profiles could be encountered in the following cases:

- A user has been cloned from a Windows NT 4.0 domain.
- A user has been cloned from a Windows 2000.
- A user has been cloned from a Windows 2000 domain, but is logging on at a Windows NT 4.0 workstation.

The following section describes some options for handling these cases.

Profile Migration

There are two basic methods of making a profile available to the migrated user, while at the same time allowing a fallback route.

- **Shared profiles** - Make the same profile available to both the user's original account and the new account. One copy of the profile is accessed and updated by both accounts
- **Copied profiles** - Copy the original profile from its current location under the key named after the user's original SID to a key named after the user's new SID. Each account is associated with its own, separate copy of the profile. Updates to one are not reflected in the other.

Both approaches have their advantages and disadvantages.

Shared Profiles

Table 7. Advantages and Disadvantages of Sharing Profiles

Advantages	Disadvantages
Updates to the profile (for instance, changes to My Documents, shortcuts, and so on) while logged onto one account are accessible when subsequently logged onto the other account	When moving between the old and new accounts, results are currently unpredictable. An example would be a new Windows 2000 account profile, perhaps containing Group Policy references, is used when falling back to a source account where different group policy (or no group policy) is deployed.
Conserve disk space as only one copy of the profile is stored	
Availability of a utility to enable profile sharing	

Copied Profile Migration

Table 8. Advantages and Disadvantages of Copying Profiles

Advantages	Disadvantages
Behavior is more predictable, and because data is not shared between the profiles, there's no chance of "polluting" the profile of one account with data appropriate only for another account in a different domain and forest.	There may be unpredictable "fallback" results with application deployment policy in the new domain. Even though the old account profile is not altered by destination account activity, applications deployed on the desktop by destination domain group policy may under certain circumstances, try to uninstall themselves or do other unpredictable things.
	Consumes extra disk space as two profiles are stored

Microsoft's Domain Migration Tool

Microsoft is working on a migration tool that will be freely downloadable from the Web. The tool will provide graphical support for the operations described, and support the migration scenarios described below.

More details about the tool will be available nearer release time.

Domain Migration Basic Utilities and Domain Migration Scenarios

We have already referred to a number of domain migration basic utilities. The basic utilities are a set of COM objects and sample scripts designed to form the basis of customer adapted administration utilities, and to support a number of domain migration scenarios that Microsoft has documented and tested. The scenarios have been developed based on feedback from customers about their migration requirements. The domain migration scenarios Microsoft has documented are focused on facilitating the initial move to Windows 2000.

Microsoft has been working with Independent Software Vendors (ISVs) to ensure that their tools support these scenarios, though their tools may enable additional approaches, and their tools may provide more sophisticated features.

Domain Migration Scenarios

Microsoft has documented two migration scenarios for Windows 2000, which will answer most customers' restructuring requirements. The scenarios are aimed at facilitating the movement of users and machines from Windows NT source domains to Windows 2000 targets. The scenarios are:

- Migrating users incrementally from Windows NT to Windows 2000
- Consolidating a resource domain into a Windows 2000 OU

Migrating Users Incrementally from Windows NT to Windows 2000

The aim of this scenario is to describe the steps and the basic utilities required for an organization to migrate its users incrementally to a pristine Windows 2000 environment without impacting its Windows NT production environment.

The implication of not impacting the current production environment is that the environment must remain untouched during the migration. A consequence of this is that during the migration the customer can always fall back to the old production domain should the need arise.

Once the migration is complete, it is envisaged that the old account domain will be decommissioned and the DCs reassigned.

Scenario Steps

The scenario steps are as follows:

1. **Create the pristine Windows 2000 forest –** This simply involves using standard procedure to create a new Windows 2000 destination forest that reflects the requirements and structure identified in the customer's namespace planning activities. Domains created in the new forest will be native mode Windows 2000 domains.
2. **Establish the trusts required to the forest to maintain resource access** – This activity involves using NETDOM to query what trusts currently exist from any resource domains to the Windows NT source domain.

The output from NETDOM can then be compared with the list of trusts it has been determined are required to enable resource access to users and groups in the target domain. NETDOM can then be used to establish any trusts that do not already exist.

3. **Clone all source global groups in the target domain** – As discussed earlier, most resources will be protected using ACLs that reference global groups, usually indirectly via shared or machine local groups. Once trusts have been established, the next task is to ensure that the relevant global groups are available in the target domain.

The simplest way to achieve this is to clone all global groups using ClonePrincipal.

4. **Identify and clone sets of users** – Once the source global groups have been cloned to the target domain, the task of migrating users can begin.

Because in most instances the customer will want to move sets of users incrementally, this will be an iterative task, involving identifying the sets of users to migrate, then using ClonePrincipal to clone the source users in the destination domain.

As part of this task, each user migrated will have to be added to the destination clone of any global groups it had membership of in the source domain.

5. **Decommission the source domain** – When all users and groups have moved permanently to the destination forest, the final task will be to decommission the source domain, which involves powering off and removing the source domain BDCs, and finally the source domain PDC.

If the intention is that these DCs are to be reassigned in the new forest, then they can be upgraded to Windows 2000 and either promoted to DCs or left as member servers as described earlier.

Each step in the scenario should be tested, particularly during the user migration phase. It might be prudent to test logon for certain users during each migration. If an error occurs at any stage prior to decommissioning, the process can be suspended and work can continue in the source production domain.

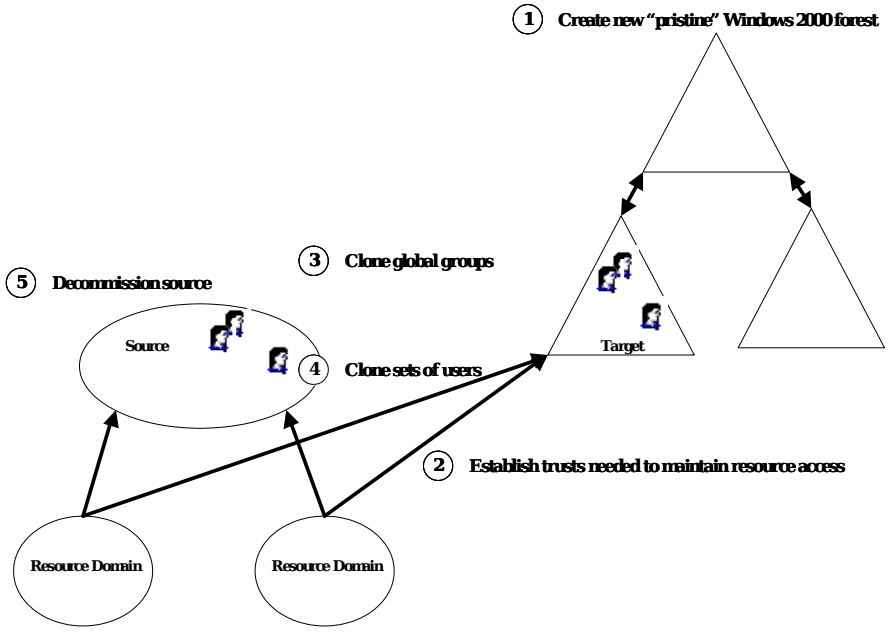


Figure 9. Migrating users incrementally from Windows NT to Windows 2000

Consolidating a Resource Domain into a Windows 2000 OU

The aim of this scenario is to describe the steps and the basic utilities required for an organization to consolidate a number of resource domains into an OU in a native mode Windows 2000 domain. Typically, the impetus for this is to reduce the costs of administering complex trusts.

As part of this scenario the application servers will become member servers in the target OU. It is assumed that the application servers in each domain will be making use of shared local groups. It is also assumed that the domains may contain some member servers and workstations.

Once the restructuring has been achieved the organization can decommission the old domains.

Scenario Steps

The scenario steps are as follows:

1. **Establish any trusts required from the target domain to account domains outside the forest** – This scenario differs from the previous one in that it is the resource domains that are being brought into the target forest and the account domains remain outside the forest, for the time being at least. The principal remains the same however, so trusts will have to be established from the target domain to the account domains to maintain resource access.

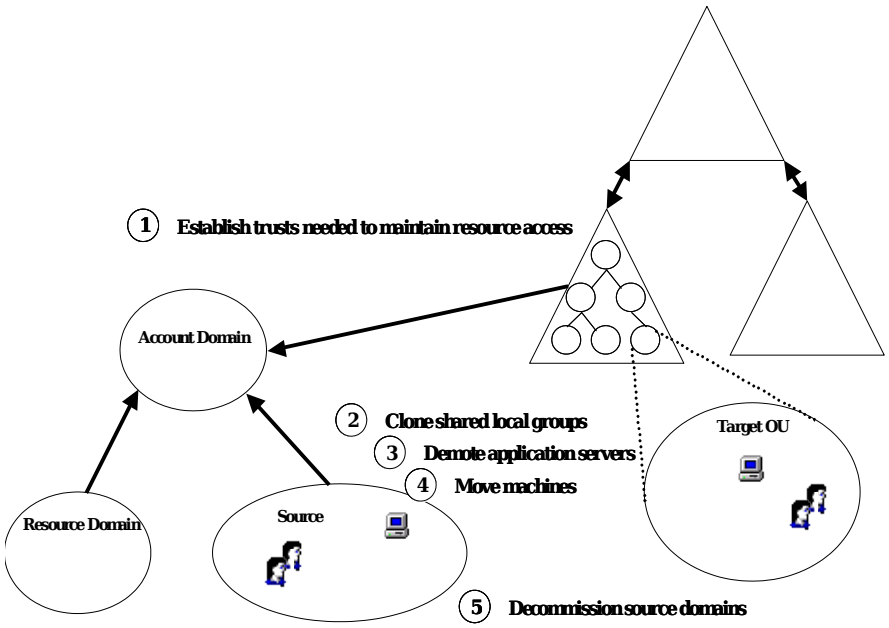
2. This activity involves using NETDOM to query what trusts currently exist from the resource domains to the account domains.
   The output from NETDOM can then be compared with the trusts that already exist from the target domain to the account domains. NETDOM can then be used to establish any trusts that do not already exist.
   Clone all shared local groups – As discussed earlier, shared local groups only have scope within the domain they were created in and are only shared between DCs in that domain. In this scenario it is not necessary to move all DCs to the target domain immediately, so to ensure that resource access is maintained while DCs and resources are split between the source and target domains it will be necessary to clone shared local groups to the target domain using ClonePrincipal.
3. **Demote application servers to member servers** – Once all the shared local groups have been cloned it is possible to start the process of converting the application servers to member servers in the target OU.

- There are currently some limitations on how BDCs can be upgraded. One of the effects of this is that there is no easy way of upgrading a BDC and demoting it to a member server unless the PDC has been previously upgraded. There are two approaches to upgrading and demoting BDCs:
  If possible it is recommended that you upgrade the PDC of the resource domain to Windows 2000 and run the domain in mixed mode during the transition period. You can now upgrade each BDC to be demoted.
- During each BDC upgrade DCPROMO will be launched and you will be given the option of making the BDC a replica in a Windows 2000 domain, or a member server in the domain. You should select the latter option, so that the machine is now a member server in the resource domain.
  If upgrading the PDC is not possible or desired, for each upgrade you will need to take the BDC offline and promote it to PDC. Once you have promoted the BDC to PDC you can then upgrade to Windows 2000, effectively making the offline DC, PDC in a clone Windows 2000 mixed mode domain. Once the PDC has been upgraded offline, you can then run DCPROMO to demote the PDC to a member server, and join it to the target domain.

4. **Move member servers (including former BDCs) and workstations** – During this step you can use NETDOM to create a computer account in a destination domain OU for the member server or workstation to be moved, and join the computer to the destination domain.
5. **Decommission the source domain** – When all groups and machines have moved permanently to the destination forest, the final task will be to decommission the source domain, which involves powering off

and removing the source domain BDCs, and finally the source domain PDC.

If the intention is that these DCs are to be reassigned in the new forest,



then they can be upgraded to Windows 2000 and either promoted to DCs or left as member servers as described earlier.

It should be noted that in this scenario, when demoting BDCs to member servers, you should move them over to the target domain as quickly as possible. Unless the domain is in native mode and shared local groups have been converted to domain local groups, resources accessible through these groups will not be available on the member servers.

Figure 10. Consolidating a resource domain into a Windows 2000 OU

The purpose of this section is to guide you through the decision points in determining your migration roadmap, now that you have an understanding of the main migration concepts.

Upgrade Decisions

1. Is upgrade appropriate for you?

You will probably answer “yes” if some or all of the following conditions are true:

- You are happy with your current domain structure.
- You are happy with much of your domain structure and can carry out a two-phase migration, where you can upgrade to Windows 2000, then restructure to fix any problems.
- You feel you can manage the migration without impacting your production environment.

2. In what order should you upgrade?
The answer depends on whether you are referring to order of upgrade of DCs or order of upgrade of the domain.

3. In what order should you upgrade DCs?
Within a domain the order of upgrade is straightforward, you need to upgrade the PDC first, but you should be aware of possible complications, such as the use of LMRepl in the domain to be upgraded, with the PDC hosting the export directory. In this case you will need to re-host the export directory before upgrading the PDC.

4. In what order should you upgrade domains?
You will generally experience more benefit by upgrading your account domains first, since you will get the most “quick wins” in terms of easier administration, and delegation.

Next you should upgrade your resource domains.

5. In what order should you upgrade server and workstations?
You can upgrade servers and workstations at any time; this is not reliant on the existence of a Windows 2000 infrastructure.

As regards member servers, if you feel that loosening the security of Active Directory to enable the use of Windows NT RAS servers would compromise your security policy, you should seek to upgrade these first.

If you are using LMRepl to replicate scripts within the domain, then you should upgrade the server hosting the export directory last.

6. When should you switch to native mode?
You should switch to native mode as soon as possible to get access to full Windows 2000 functionality, such as better directory scalability, universal and domain local groups, and group nesting.

You cannot switch to native mode until all the DCs in a domain have been upgraded.

Restructure Decisions

1. Do you need to restructure?
You will probably answer “yes” if some or all of the following conditions are true:

- You are happy with much of your domain structure and can carry out a two-phase migration, where you can upgrade to Windows 2000, then restructure to fix any problems.
- You are unhappy with your current domain structure.
- You feel you cannot manage the migration without impacting your production environment.

1. When should you restructure?
The answer to this question very much depends on the reason you are

restructuring (see above question).

If you can solve your migration requirements by doing a two-phase migration, then you should restructure after upgrade.

If you feel that your domain structure is for some reason unsalvageable, or that you cannot avoid impacting your production environment, you should restructure from the start.

Migration from Windows NT 4.0 to Windows 2000 Active Directory can be achieved by upgrade or restructure.

Microsoft has provided an easy, low-risk migration path in the form of domain upgrade. Many customers will find that this answers all of their migration requirements. Microsoft has listened to customer feedback and is also committed to providing great tools to support domain restructure, either as part of the initial migration, or post-migration to support corporate operations such as acquisition and divestiture. The Microsoft migration tool and the basic utilities represent the first step.

Microsoft tools and utilities represent only part of the story. Many customers will have strong working relationships with ISV tool vendors in this area, and will want to continue using tools they are familiar with. Microsoft has been working with ISVs to ensure that their domain restructuring tools are able to make use of features such as SIDhistory and cloning operations.

For More Information

For the latest information on Windows 2000 Server, visit the Web site at http://www.microsoft.com/windows2000/default.asp.